

(19)日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開平 1 0 - 2 6 2 2 1 0

(43)公開日 平成 1 0 年 (1 9 9 8) 9 月 2 9 日

(51)Int.Cl.	識別記号	庁内整理番号	F I	技術表示箇所
H04N 5/91			H04N 5/91	N
G11B 27/00			G11B 27/00	E
H04N 5/76			H04N 5/76	B
			G11B 27/00	E

審査請求 未請求 請求項の数 6 O L (全 2 2 頁)

(21)出願番号 特願平 9 - 3 1 3 2 8 6

(22)出願日 平成 9 年 (1 9 9 7) 1 1 月 1 4 日

(31)優先権主張番号 6 0 / 0 3 0 9 5 5

(32)優先日 1 9 9 6 年 1 1 月 1 5 日

(33)優先権主張国 米国 (U S)

(71)出願人 0 0 0 0 0 5 1 0 8

株式会社日立製作所

東京都千代田区神田駿河台四丁目 6 番地

(71)出願人 5 9 7 1 6 0 2 6 6

フューチャーテル・インコーポレイテッド
アメリカ合衆国、9 4 0 8 6 カリフォル
ニア州、サニーヴェイル、イースト・アー
クス・アヴェニュー 1 0 9 2

(72)発明者 エリック・ティー・ブリューワー

アメリカ合衆国、9 5 0 7 0 カリフォル
ニア州、サラトガ、パロ・オークス・コー
ト 1 8 9 9 4

(74)代理人 弁理士 筒井 大和

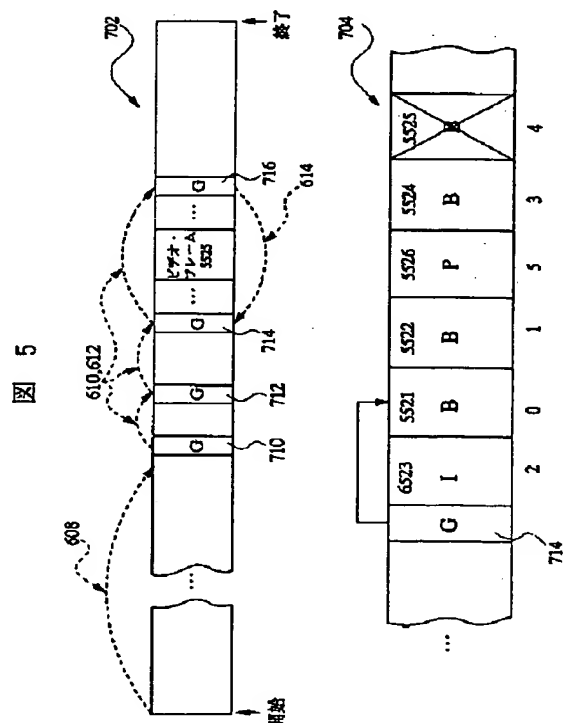
最終頁に続く

(54)【発明の名称】 オーディオビジュアル・ファイル内部での検索方法および検索装置

(57)【要約】

【課題】 オーディオビジュアル・ファイル内部で検索を行なうための方法および装置を提供する。

【解決手段】 検索エンジンは、ビデオ・ファイルの中の最後のGOPヘッダ716を識別し、次にファイルの中の最後のビデオ・フレームを識別して、ビデオ・ファイル702の中のフレームの数を正確に判別することができる。ファイルの中の最後のビデオフレーム(P-5526)は、各ビデオ・フレームの画像ヘッダ(すなわち最後のGOPの内部の画像ヘッダ)を識別し、どの画像ヘッダがより大きい時間基準フレーム番号を示すかを判別することによって識別される。すなわち、最も大きい時間番号をもつビデオ・フレームが、ビデオ・ファイルの中の最後のビデオ・フレームである。さらに、検索エンジンは、識別されたビデオ・フレームと最も近いオーディオ・フレームを関連づけてオーディオ・ビデオ検索を完了することができる。



【特許請求の範囲】

【請求項 1】 オーディオビジュアル・ファイルのバイトサイズと、前記オーディオビジュアル・ファイルの速度情報とに基づいて、前記オーディオビジュアル・ファイルに含まれる全フレーム数を推定する第 1 のステップと、

前記全フレーム数と、前記オーディオビジュアル・ファイルのバイトサイズと、前記オーディオビジュアル・ファイルの前記速度情報と、フレーム番号とに基づいて、検索対象の目的のフレームの位置をバイト数で求める第 2 のステップと、

前記バイト数に基づいて、前記オーディオビジュアル・ファイル内の検索対象の目的の前記フレームの近傍にアクセス開始位置を位置付ける第 3 のステップと、

を含むことを特徴とするオーディオビジュアル・ファイル内部での検索方法。

【請求項 2】 請求項 1 記載のオーディオビジュアル・ファイル内部での検索方法において、

前記オーディオビジュアル・ファイルの前記速度情報は、前記オーディオビジュアル・ファイルの再生時のビット・レート、およびフレーム・レートからなることを特徴とするオーディオビジュアル・ファイル内部での検索方法。

【請求項 3】 請求項 1 記載のオーディオビジュアル・ファイル内部での検索方法において、

前記第 3 のステップでは、前記第 2 のステップで得られた前記バイト数の位置から、所定の時間分のバイト数だけ前記オーディオビジュアル・ファイルの先頭側に逆上った位置に前記アクセス位置を位置付け、当該アクセス開始位置以降に存在する各フレームのフレーム番号と、前記検索対象の目的の前記フレーム番号の大小関係を判定する操作を各フレーム毎に反復して、目的の前記フレーム番号の前記フレームの先頭部分にアクセス開始位置を位置付けることを特徴とするオーディオビジュアル・ファイル内部での検索方法。

【請求項 4】 編集対象のオーディオビジュアル・ファイルが格納される第 1 の記憶媒体と、

オーディオビジュアル・ファイルのバイトサイズと、前記オーディオビジュアル・ファイルの速度情報とに基づいて、前記オーディオビジュアル・ファイルに含まれる全フレーム数を推定する第 1 のステップと、前記全フレーム数と、前記オーディオビジュアル・ファイルのバイトサイズと、前記オーディオビジュアル・ファイルの前記速度情報と、フレーム番号とに基づいて、検索対象の目的のフレームの位置をバイト数で求める第 2 のステップと、前記バイト数に基づいて、前記オーディオビジュアル・ファイル内の検索対象の目的の前記フレームの近傍にアクセス開始位置を位置付ける第 3 のステップと、
よって、前記記憶媒体の前記オーディオビジュアル・ファイル内部の任意の編集対象データを選択的に切り出

して任意の第 2 の記憶媒体に保管する操作を行う検索制御論理と、

前記第 2 の記憶媒体に保管された前記編集対象データを読出して、任意の前記オーディオビジュアル・ファイルの一部に組み込んで出力する操作を行う編集制御論理と、

を含むことを特徴とするオーディオビジュアル・ファイル内部での検索装置。

【請求項 5】 請求項 4 記載のオーディオビジュアル・ファイル内部での検索装置において、

前記検索制御論理は、

前記オーディオビジュアル・ファイルが、それ自体で独立して復号可能な第 1 のフレームと、過去のフレームのデータに依存する第 2 のフレームと、過去および未来のフレームの双方のデータに依存する第 3 のフレームとを含む場合、

前記編集対象データの先頭部分の前記フレームが過去または過去および未来のフレームのデータに依存する前記第 2 または第 3 のフレームである時、より過去のフレームのデータを含めた復号化および前記第 1 のフレームへの再符号化を行う第 1 の操作、

前記編集対象データの後端部分が、前記フレームが未来のフレームのデータに依存する第 3 のフレームである時、より未来の前記フレームのデータを含めた復号化および前記第 1 または第 2 のフレームへの再符号化を行う第 2 の操作、

の少なくとも一方の操作を実行することを特徴とするオーディオビジュアル・ファイル内部での検索装置。

【請求項 6】 請求項 4 または 5 記載のオーディオビジュアル・ファイル内部での検索装置において、

前記検索制御論理および前記編集制御論理は、外部から与えられる編集制御情報が格納された編集リストに基づいて、任意のオーディオビジュアル・ファイルからの前記編集対象データの切り出し操作および組み込み操作を自動的に実行する機能を備えたことを特徴とするオーディオビジュアル・ファイル内部での検索装置。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、広くは、ビデオ・ファイルの編集の方法に関する。より詳しくは、本発明は、多重オーディオビジュアル・ファイル内部でのあらかじめ定められたビデオ・フレームの検索の各種方法および装置に関する。一側面にあつては、ファイル・バイト・オフセットを基準にして、ターゲット・フレームのデータを検索してアクセスする方法および装置が開示される。

【0002】

【従来の技術】MPEG（動画エキスパート・グループ）は、国際規格機構（ISO）が定めたデジタル・ビデオおよびオーディオ信号を簡潔にあらわすための構文

法に関する規格である。この構文法では、一般に、ビット・ストリームを符号化するときにしたがうべき規則の数を最小限にしてしかも受信したビット・ストリームを明確に復号できることが求められる。当業者には周知のように、ビット・ストリームは、ビデオおよびオーディオ構成要素に加えて「システム」構成要素を含むものである。一般に、システム構成要素は、各ビデオおよびオーディオ構成要素を組み合わせ同期させて単一のビット・ストリームにするために必要な情報を含んでいる。具体的には、システム構成要素によって復号器でのオーディオ/ビデオの同期が可能となる。

【0003】MPEG-1と呼ばれる最初のMPEGが公表された後、MPEG-2で知られる次のMPEGが導入された。一般的にいて、MPEG-2は、放送されるビデオをより効率的にあらわすことができる改良された構文法を規定したものである。成立の経緯から、MPEG-1では、1.5Mビット/秒の速度でデータを取り扱い、各フレームが352画素×240ラインの解像度のものを毎秒約30ビデオ・フレーム(NTSC)または各フレームが352画素×288ラインの解像度のものを毎秒約25ビデオ・フレーム(PAL)再構成する場合に最適の結果が得られるものとなっている。したがって、復号されたMPEG-1ビデオは、消費者のビデオテープ(VHS)の知覚される品質にほぼ近似している。それに対して、MPEG-2は、4.0から8.0Mビット/秒のデータ速度でのCCIR601の解像度をあらわした720画素×480ライン(NTSC)または720画素×576ラインのフレーム解像度をあたえるように設定されている。以下、簡単のために、これら二つのMPEG規格の間の相違が問題となる場合をのぞいて、「MPEG」という用語は、現在規定されているおよび将来規定されるであろうビデオおよびオーディオ符号化および復号アルゴリズムをさすものとする。

【0004】通常、復号の過程は、ビデオ、オーディオ、およびシステム情報を含むMPEGビット・ストリームが符号化された個別のビデオおよびオーディオ・ビット・ストリームを生成する役割りを果たすシステム復号器によってデマルチプレクスされるときに始まる。なお、これらの符号化されたビデオおよびオーディオ・ビット・ストリームは、その後、ビデオ復号器およびオーディオ復号器で復号することができる。現在、符号化されたビデオ・ビット・ストリームの構造に関心が向けられている。一般に、符号化されたMPEGビデオ・ビット・ストリームは、明確なデータ構造階層に組織される。この階層の最も高いレベルには「ビデオ・シーケンス」がある。これは、シーケンスヘッダ、1以上の画像グループ(GOP)、およびシーケンスの終わりコードを含むものとして行うことができる。GOPは、ビデオ・シーケンスのサブセットであり、各GOPは、1以上の画像を含むことができる。以下に説明するように、GOP

は、それによってビデオ・シーケンスのある画定されたセグメントにアクセスできるためきわめて重要である。ただし、GOPは、きわめて大きくなる場合がある。

【0005】1つのGOP内部の各画像は、左から右へまた上から下へ向けて画定されるいくつかの「スライス」に仕切られる。個々のスライスは、16×16画素の正方形の面積を占める1以上のマクロブロックで構成される。MPEG規格に記されているように、1つのマクロブロックは、四つの8×8画素「ルミナンス(輝度)」構成要素と二つの8×8「クロミナンス(色差)」構成要素(すなわち、クロマ赤およびクロマ青)を含む。

【0006】1つのGOP内部の画像の間では、画素情報の多くが類似しているかまたは同一であるため、MPEG規格は、この時間的冗長性を利用し、特定の基準画像から互いに異なる選ばれた画像をあらわすようにしている。MPEG規格は、大きく、三種類の符号化された画像フレームを定義している。第1の種類のフレームは、イントラ・フレーム(I-フレーム)である。I-フレームは、フレーム自身に含まれる情報を用いて符号化され、以前のまたは将来のフレームに含まれる情報には依存しない。その結果、I-フレームは、一般に、フレームのシーケンスの中の特定のGOPの起点を定義する。

【0007】第二の種類のフレームは、予測フレーム(P-フレーム)である。P-フレームは、一般に、前のIまたはP-フレームに含まれる情報を用いて符号化される。当業者には周知のように、P-フレームは、前方予測フレームと呼ばれる。第三の種類のフレームは、双方向フレーム(B-フレーム)である。B-フレームは、過去および未来のフレームの両方に含まれる情報にもとづいて符号化され、したがって双方向予測フレームと呼ばれる。したがって、B-フレームは、I-フレームおよびP-フレームより圧縮されたものとなる。MPEG規格は、IまたはP-フレームの間に特定の数のB-フレームを配置することを求めているが、大部分の符号器は、IおよびP-フレームの間に二つのB-フレームを選択する。このような選択の設定は、符号器の中のメモリの量および符号化される材料に必要な特性および定義などの各種要因にもとづいて行なわれている。

【0008】

【発明が解決しようとする課題】MPEG規格は、ビデオおよびオーディオ・ビット・ストリームを簡潔に符号化するための便利な構文法を定めているが、符号化されたビット・ストリームのセグメントを切り取って新しいビット・ストリームで使用する場合にはかなりの困難が生じる。とくに、P-フレームは、ビット・ストリーム内の以前のフレームからの情報を使用し、またB-フレームは、以前と将来のフレームの両方からの情報を使用するため、切り取りは、I-フレームで行なわなければ

ならない。すなわち、切り取られたセグメントは、その切り取られたセグメントの中に、I-フレームを開始フレームとしてまたPまたはI-フレームを最終フレームとしてもっていなければならない。したがって、I-フレームで切り取りを行なうと、最初のビット・ストリームに含まれる基準フレームなしには復号することができない開始および終了フレームを有するビデオ・クリップが排除されることになる。

【0009】残念ながら、通常の符号化されたビデオ・ビット・ストリームは、I-フレームの間に多数のPおよびB-フレームをもっている。その結果、切り取りが行なわれる場所が限定される不具合が生じ、符号化されたMPEGビット・ストリームは、フレームの精密さが求められるビデオ編集業に適さないものになってしまう。

【0010】従来の編集エンジンに関連する他の不具合は、ターゲット・ビデオ・フレームの検索を行なうためには、ファイル内の各フレームを時間をかけて読み取って復号する必要があることである。すなわち、特定のビデオ・フレームの検索を行なう前に、編集プログラムがファイル内の各ビデオ・フレームを読み取って復号し、各フレームの時間的基準を判別する必要がある。各フレームが読み取られて復号されれば、ターゲット・フレームの検索を行なうことができる。残念ながら、大多数のビデオ・ファイルは、きわめて大きいものである。例えば、3時間のビデオ・ファイルは、フレーム・レートが毎秒30フレームとすれば、約324000ものビデオ・フレームを含むものとなる。ターゲット・フレームの検索を行なう前に324000ものビデオ・フレームを各々読み取って復号するとすれば、それは、きわめて手間のかかる今日のビデオ編集作業には不適なものとなることは理解されよう。さらに、従来の検索アルゴリズムは、ビデオ・ファイル内のフレームの正確な数を確認する前にビデオ・ファイルを読み取って復号する必要がある。

【0011】以上の説明から、手間をかけて予めビデオ・ファイルの各フレームを読み取って復号することを必要とせずに、1つのビデオ・ファイル内部のターゲット・ビデオ・フレームの検索を効率的に行なうための方法および装置が求められている。

【0012】

【課題を解決するための手段】本発明の目的にもとづいて上の課題を解決するために、1つのオーディオビジュアル・ファイル内部のターゲット・ビデオ・フレームの検索を高速かつ効率的に行なうための方法および装置が開示される。1つの実施の形態にあつては、本発明の検索装置は、まず、オーディオビジュアル・ファイル内部のターゲット・フレームのバイトであらわした推定位置を判別する。バイトであらわした推定位置が判別されると、該推定位置からバイトであらわしたあらかじめ定め

られた秒数が差し引かれ、推定時間位置が示される。次に、検索装置は、ターゲット・ビデオ・フレームの少なくとも1つの画像グループ(GOP)ヘッダだけ前にある推定時間位置まで飛び越しを行なう。

【0013】ここで、検索エンジンは、推定時間位置の前にある任意のGOPヘッダへ進む。各GOPヘッダで、検索エンジンは、そのGOPヘッダの時間コードから得られるフレーム番号がターゲット・フレームの番号より大きいかな否かを判別する。GOPヘッダの時間コードから得られるフレーム番号がターゲット・フレームの番号より大きい場合には、検索エンジンは、前に読み取られて保管されたGOPヘッダへ逆戻りする。この時点で、前のGOPヘッダは、ターゲット・フレームを含んでいることが好ましい。ターゲット・フレームを識別するために、検索エンジンは、前のGOPヘッダの時間コードから得られたフレーム番号をターゲット・フレーム番号から差し引いて、GOPヘッダ内部にターゲットの時間基準フレーム番号を生成する。これで検索エンジンは、ターゲット時間基準フレーム番号まで移動してターゲット・フレームの特定を行なうことになる。

【0014】検索エンジンがターゲット・フレームを識別したら、そのターゲット・フレームに関するファイル・バイト・オフセットがわかって記憶される。この記憶されたバイト・ファイル・オフセットを用いれば、任意のときにターゲット・フレームの検索を行ない、ターゲット・ビデオ・フレームに含まれるデータにアクセスすることができる。

【0015】他の一実施の形態にあつては、オーディオビジュアル・ファイルの中のビデオ・フレームの数を正確に判別することのできる検索エンジンが開示される。最初、検索エンジンは、オーディオビジュアル・ファイルの終わりを識別してそこへ移動する。終わりに達したら、検索エンジンは、あらかじめ定められた時間だけビデオ・ファイルの中のある位置まで逆戻りする。次に、検索エンジンは、時間的にファイルの前方へ移動して任意のGOPヘッダを識別し、任意の識別されたGOPヘッダが保管される。検索エンジンは、ファイルの終わりに達するまで時間的に前方へ移動しながらGOPヘッダの識別と保管を続ける。終わりに達したら、検索エンジンは、前に保管したGOPヘッダへ逆戻りする。前に保管したヘッダに達したら、検索エンジンは、該前に保管したGOPヘッダに関連するあらかじめ定められた数のビデオ・フレーム画像ヘッダの各々を読み取って各ビデオ・フレームごとの時間基準フレーム番号を判別する。あらかじめ定められた数のビデオ・フレーム画像ヘッダの各々が読み取られるのに応じて、前の時間基準フレーム番号が現在の時間基準フレーム番号より大きいかな否かが判別される。

【0016】検索エンジンは、この判別にもとづいて、前の時間基準フレーム番号より小さい時間基準フレーム

番号を有するすべてのビデオ・フレームを無視する。ファイルの終わりに達した時点で、ビデオ・ファイルの中の（確認可能なフレーム番号をもつ）最後のフレームが、前に保管されたGOPヘッダの中の最も大きい時間基準フレーム番号をもつフレームとして識別される。検索エンジンは、ファイルの中の各フレームを手間をかけて読み取りまた復号する必要なしに、ビデオ・ファイルの中のビデオ・フレームの数を正確に判別することができる。

【0017】本発明は、数多くの効果をもたらすが、とくに大きな効果は、ターゲット・ビデオ・フレームを検索する前あるいはオーディオビジュアル・ファイルのビデオ・フレームの数を判別する前に、手間をかけてオーディオビジュアル・ファイルの中の各ビデオ・フレームを読み取り、復号し、指標付けを行なう必要がないことである。

【0018】

【発明の実施の形態】本発明ならびにその効果は、添付の図面を参照して行なう以下の説明から最もよく理解されよう。

【0019】本発明は、広くは、オーディオビジュアル・ファイル内部での検索の方法および装置を開示するものである。検索エンジンは、手間をかけてオーディオビジュアル・ファイルの中の各ビデオ・フレームを読み取り、復号し、指標付けを行なう必要なしにターゲット・ビデオ・フレームを効率的に検索するように実装される。さらに、検索エンジンは、1つのビデオ・ファイルの中の最後のGOPヘッダを識別し、さらに該ファイルの中の最後のビデオ・フレームを識別することによって、該ビデオ・ファイルの中のフレーム数を正確に判別することができる。ファイルの中の最後のビデオ・フレームは、各ビデオ・フレームの（すなわち、最後のGOP内部の）画像ヘッダを読み取り、どの画像ヘッダがより大きい時間基準フレーム番号を示すかを判別することによって識別される。すなわち、最大の時間番号をもつビデオ・フレームは、該ビデオ・ファイルの中の最後のビデオ・フレームである。さらに、検索エンジンは、識別されたビデオ・フレームを最も近いオーディオ・フレームと関係づけてオーディオビデオ検索を完了することができる。

【0020】説明をわかりやすくするために、本発明の検索エンジンを、ビデオ・ファイルの編集との関連で説明する。ビデオの編集では、一般的に、オーディオビジュアル・ファイル内部の編集されたビデオ・セグメントの境界を画定するためにターゲット・ビデオ・フレームの検索を行なう必要がある。したがって、フレームに関して正確かつ効率的に検索を行なう方法および装置の必要性を理解するために、ビデオ・ファイルの編集の各種の方法を概観しておくことが有用であろう。「切り取り、コピー、マークイン、マークアウト」などの用語

は、すべて、特定のビデオ・フレームを識別するための検索エンジンの実施の形態に関して用いられることを理解されたい。ターゲット・フレームが識別されたら、該ターゲット・フレームのデータの内容に高速でアクセスすることができる。

【0021】本発明の一実施の形態にあつては、MP EGビット・ストリーム・ファイルからビデオのセグメントを切り取り、切り取られたセグメントの部分を処理して元のビット・ストリーム・ファイルに含まれる情報から独立した1つのビット・ストリーム・セグメントを生成するための方法が開示される。一般に、編集エンジンは、特定の編集作業を要求するアプリケーションに応じて提供されるオペレータ（演算子）の編集リストを使って二つの処理パスで独立のセグメントを生成するために編集対象セグメントを処理する。最初の処理パスでは、編集エンジンは、好ましくは切り取られたセグメントの始めと終わりにあるフレームの種類にもとづいて、該編集対象セグメント用のグル（のり付け）セグメントを生成する。第二の処理パスでは、最初のパスで生成されたすべてのグル・セグメントが編集対象セグメントの未処理部分に貼り付けられる。すべてのグル・セグメントと未処理部分が正しい時間順序で互いに貼り付けられると、貼り付けられたセグメントがアプリケーションへ送られる。貼り付けられたセグメントは、ビデオ・フレームを正確に復号するための最初のビット・ストリーム・ファイルの中に含まれている情報を必要としない。

【0022】図1は、ソース・ファイルに含まれるフレーム情報から独立したビデオ・フレーム・セグメントを生成することに関係する処理ステップを説明するために用いる多くのビデオ・フレームのシーケンスを示す図である。例として、フレームがMP EG規格のフォーマットにもとづいて処理された後で符号化される順序を示すビデオ・フレームの符号化の順序のストリーム50が示されている。例として示したこの符号化の順序のストリーム50では、最初のフレームは、I-フレームであり、その後にP-フレーム、B-フレーム、B-フレーム、P-フレーム、B-フレーム、B-フレーム等々が続く。本発明の編集アルゴリズムは、任意の適当に配列されたフレームのシーケンスを処理することができるが、表示の順序でフレームのシーケンスを処理することが好ましい。

【0023】すなわち、フレーム0からフレーム36まで時間順序で配列されたフレーム・ストリームは、表示順序ストリーム52の中で処理されるフレームの順序を識別する。比較のために、符号化順序ストリーム50の中のフレームの対応する時間順序を対応するフレームの下に示してある。もちろん、表示順序ストリーム52は、単に例であり、本発明にもとづいて他の適当な表示順序ストリームを適当に処理できることは理解されよう。

【0024】ビデオ・フレームのセグメントが表示順序ストリーム52から切り取られると、マークインの位置およびマークアウトの位置が設定されて、切り取られるフレームの番号がマークされる。例として、マークインの位置がP-フレームであるフレーム9に設定され、マークアウトの位置がB-フレームであるフレーム28に設定されたとする。したがって、表示順序ストリーム52から切り取られるフレームのセグメントは、フレーム9から28までとなる。切り取りの大きさがきまると、フレーム9がI-フレームでなくフレーム28がI-フレームまたはP-フレームでない場合には、切り取りの始めに、フレームは、P-フレーム9では過去のフレームからの情報を、またB-フレーム28では過去および未来のフレームからの情報を必要とする。その結果、フレーム9から14およびフレーム25から28は、予測型のフレームであり、表示順序ストリーム52に残っているフレームから十分な情報(コンテキスト)を把握しないと復号することができないことになる。

【0025】ビデオ・フレームの切り取られた全セグメントを復号可能にするために、ビデオ・フレーム9から14および25から28は、該切り取られた全セグメントを復号可能でかつ最初の表示順序ストリーム52に含まれる情報から独立したものとする処理を受ける。例として、フレーム9から14および25から28が復号されてI-フレームに再符号化された後に「ドラフト・モード」で処理されたセグメント54を示す。便宜上、処理されたフレーム9から14を「イン・グルセグメント」と呼び、フレーム24から28を「アウト・グルセグメント」と呼ぶ。さらに、未処理のフレーム15から23を「ミドル・グルセグメント」と呼ぶ。

【0026】本実施の形態にあつては、イン・グルおよびアウト・グルセグメントは、フレーム9より前のリファレンスフレームを編集対象セグメント内から削除したI-フレームに符号化されており、またB-フレーム25、26、27、および28も、フレーム28より後のフレームに含まれる情報を必要としない。再符号化されたイン・グルおよびアウト・グルセグメントも、IおよびPの組み合わせに符号化することができ、I-フレームは、切り取らとられた「I-Pモード」セグメント56に示すイン・グルセグメントおよびアウト・グルセグメントの両方の始まりとなることができることは理解されよう。さらの他の一実施の形態にあつては、再符号化されたイン・グルおよびアウト・グルセグメントは、I-P-Bフレームに符号化して、I-フレームが、切り取られた「I-P-Bモード」セグメント58に示すイン・グルセグメントおよびアウト・グルセグメントの両方の始まりとなることができる。セグメント58が実施される場合には、P-フレーム間の距離(すなわち、フレームの数)を判別することが好ましい。さらに、上に述べた各モードのGOPの大きさも判別することが好ま

しい。

【0027】以下に詳細に説明するように、マークインとマークアウトの位置の間の切り取るセグメントが選定されたら、マークインの位置がすでにI-フレームでなければ、マークインの位置から最も前のI-フレームが識別される。この例では、フレーム6が、表示順序ストリーム52の最も前のI-フレームである。このようにして、MPEG復号器は、I-フレーム6を復号して、マークイン・フレーム-9を含みまたミドル・インフレーム15の前の1フレームまで伸びるイン・グルセグメントの中でフレームを復号し再符号化するのに十分な情報を獲得することができる。例として、復号器が、I-フレーム6を画素ビットマップに復号して情報を獲得すると、処理が進行してフレーム9、10、11、12、13、および14を復号し再符号化するように構成することができる。同様に、復号器はI-フレーム24から十分な情報を獲得しているため、フレーム25から28も、個々に復号された再符号化される。したがって、フレーム25、26、27、および28も、再符号化されて、I-フレーム24で始まる適当なアウト・グルセグメントを生成する。

【0028】図2は、本発明の一実施の形態にもとづくビデオ・ファイルの編集に用いられるデータ・フロー・アーキテクチャ100を示す。図示のように、ファイルのオーディオ構成要素の編集にも同様なアーキテクチャ(例、かげになって隠されている部分)が用いられる。

【0029】図示の実施の形態にあつては、データ・フロー・アーキテクチャ100は、多くの編集作業を行なうことのできる編集エンジン102(例、MEDIT編集エンジン)によって駆動されることが好ましい。例として、この種の作業として、ソースまたは入力ストリームからのセグメントが他のファイルで使用するためにコピーされる必要があることを要求するコピー操作を挙げることができる。他の適当な編集作業としては、フェード操作、ブレンド操作、モーフィング(形付け)操作、ティルティング(傾け)操作、テキスト・アノテーション(注釈付け)操作などを挙げることができる。一般に、MEDITエンジン102は、編集作業を要求するアプリケーションが提供するオペレータの種類に応じて異なる多くの編集作業を管理することのできるダイナミックなエンジンである。したがって、MEDITエンジン102は、複雑高度な編集作業を必要とする将来のアプリケーションが提供するオペレータを含む多数のオペレータの種類をすべて管理することができることが理解されよう。

【0030】以下では、ソース・ファイルからビデオのセグメントをコピーするなどの編集作業を行なう場合にMEDITエンジン102がたどる処理ステップの概要を説明する。一般に、コピー操作は、アプリケーション

1 0 6 がコピー操作を行なうことを要求したときに開始される。

【0 0 3 1】最初、アプリケーション 1 0 6 は、MED I T エンジン 1 0 2 に、編集する型を要求するためのチャンネル番号を識別する複数の「チャンネル・オペレータ」1 1 0、アプリケーション 1 0 6 が要求する編集機能の種類を識別する「機能オペレータ」1 1 2、および編集要求の終わりを識別する「終端オペレータ」1 1 4 を含む適当な編集リスト 1 0 8 を提供する。図示の実施の形態では、機能オペレータ 1 1 2 は、「コピー」要求を識別する。例として、機能オペレータ 1 1 2 で識別された最初のコピー要求は、チャンネル 1 である A、M P E G と呼ばれるファイルの中のフレーム 9 から 2 8 をコピーする要求であるとする。図示のように、チャンネル N である B、M P E G と呼ばれるファイルの中のフレーム 1 0 から 5 0 をコピーする要求に至るまで、他にも多くのコピーの要求があり得る。

【0 0 3 2】MED I T エンジン 1 0 2 が編集リスト 1 0 8 を受け取ると、コピーの要求が編集リスト 1 0 8 を通る二つの識別可能なパスで処理される。最初のパスでは、MED I T エンジン 1 0 2 は、編集リスト 1 0 8 全体を通読してコピーされた各セグメントのためにイン・グルまたはアウト・グルセグメントが必要か否かを識別する。もちろん、マークインおよびマークアウト・フレームがともに I - フレームであれば、イン・グルまたはアウト・グルセグメントは必要ではない。しかし、マークイン・フレームが I - フレームでないか、あるいはマークアウト・フレームが P または I - フレームでない場合には、そのコピーされたセグメントのためにグル・セグメントが生成されることになる。コピーされたセグメントのためにグル・セグメントが生成される場合には、該グル・セグメントは、適当な記憶媒体 1 4 0 に記憶される。記憶媒体 1 4 0 は、キャッシュ・メモリ、コンピューター・ハード・ドライブ、フロッピー・ディスク、あるいは適当なネットワークによって接続されて遠隔に配置された記憶媒体など任意の適当な記憶媒体とすることができる。

【0 0 3 3】第二のパスでは、MED I T エンジン 1 0 2 は、MED I T エンジン 1 0 2 によって生成される複数のスティッチャ・オブジェクト 1 4 7、1 4 8 を用いてグル（のり）部分を未処理のコピーされたセグメント（すなわち、ミドル・グル）と接合することで前に生成したグル・セグメントを利用する。以下により詳細に説明するように、スティッチャ・オブジェクトは、編集リスト 1 0 8 の各チャンネルのために生成され、特定のチャンネルに関連して生成された各スティッチャ・オブジェクトは、編集リスト 1 0 8 全体をみてそれ自身のチャンネルのためにグル・セグメントを接合する（例、他のチャンネルに関連する情報は無視する）能力をもつ。

【0 0 3 4】このようにして、編集リスト 1 0 8 の中で

識別された各チャンネルのために多数のスティッチャ・オブジェクトを生成することができる。特定の一実施の形態にあつては、各スティッチャ・オブジェクトは、正しい時系列で該特定のグル・セグメントを接合し、生成された各セグメントが適当な表示順序ストリームを生成するようにタイムスタンプを押す能力をもつ。さらに、生成された各スティッチャ・オブジェクトは、グル・オブジェクト 1 3 0 および 1 3 1 などのグル・オブジェクトを用いて、前に生成されたイン・グルまたはアウト・グルファイルからグル・セグメントを引き出すか、あるいはミドル・グルセグメントの位置を識別するポイントを用いて元のファイルからミドル・グルを引き出す。図 1 は、好ましくはフレーム 1 5 から 2 3 を含むミドル・グルセグメントの例を示している。貼り付けられたフレーム・データがプログラム要素ストリーム（P E S）としてマルチプレクサ 1 5 0 に出力されると、マルチプレクサ 1 5 0 は、生成されたすべてのスティッチャ・オブジェクトから P E S データを引き出し、コピーされたセグメントを MED I T 1 0 2 を介してアプリケーション 1 0 6 へ出力する。

【0 0 3 5】図 2 の全体のデータの流れを説明するために、アプリケーション 1 0 6 が、チャンネル 1 から A、M P E G ファイル 1 2 4（すなわち、図 1 の表示順序ストリーム 5 2）からフレーム 9 から 2 8 をコピーする操作を要求する場合を仮定する。MED I T エンジン 1 0 2 は、最初のパスの間に全編集リスト 1 0 8 を通読して、前の編集要求の間にグル・セグメントがすでに生成されてグル・ファイルの中に記憶されているか否かを判別する。A、M P E G ファイル 1 2 4 からのフレーム 9 から 2 8 のコピー操作のためにすでに存在するグル・セグメントはないと仮定すると、MED I T エンジン 1 0 2 は、制御オブジェクト 1 1 1（例、ダイレクト・イン・オブジェクト）を生成するコピー・オペレータ 1 0 4 を生成する。

【0 0 3 6】この実施の形態にあつては、制御オブジェクト 1 1 1 は、検索エンジン 1 1 8 を使用して、A、M P E G ファイル 1 2 4 の中でコピーするために識別された適当なビデオ・フレームを検索する。特定のターゲット・ビデオ・フレームを検索することに関係するアルゴリズムは、図 3 から図 8 を参照して以下にさらに詳細に説明する。適当なフレームが検索されると、復号器は、最も前の I - フレーム 6 を復号して、復号器 1 2 0 にイン・グルセグメント内部のフレームを処理するための適当な情報（コンテキスト）を提供する。復号器 1 2 0 が適当な情報を獲得すると、フレーム 9 は、復号器 1 2 0 によって復号され、画素ビットマップに変換され、これがコピー・オペレータ 1 0 4 へ送られる。

【0 0 3 7】コピー・オペレータ 1 0 4 は、ビットマップ情報を、コピー・オペレータ 1 0 4 によって生成し符号器 1 1 5 を有する制御オブジェクト 1 1 3（例、ダイ

10

20

30

40

50

レクターアウト オブジェクト)へ送る。符号器 1 1 5 は、再符号化されたフレームを A. M P E G グル・ファイル 1 2 6 の中に記憶しているグル・オブジェクト 1 1 6 を呼び出す。図示のように、A. M P E G グル・ファイル 1 2 6 は、キャッシュ・メモリなどの記憶媒体 1 4 0 の中に記憶されている。フレーム 9 が I フレームに再符号化されると、フレーム 1 0 から 1 4 も同様に再符号化され、A. M P E G グル・ファイル 1 2 6 などの「イン・グル」ファイルを生成する。

【0 0 3 8】M E D I T エンジン 1 0 2 は、通常、編集リスト 1 0 8 の中の各コピー要求のために個別のコピー・オペレータを生成することは理解されよう。したがって、編集リストの中の第二のコピー操作要求（すなわち、B. M P E G ファイル、チャンネル N からのフレーム 1 0 から 5 0）は、個別のコピー・オペレータ 1 0 4 によって処理され、これらのコピー・オペレータが、それ自身の検索および復号機能のために新しい制御オブジェクト 1 1 1 を生成し、また生成されたグル・フレームを符号化しておそらくは記憶媒体 1 4 0 に記憶されている他のグル・ファイルへ転送するための新しい制御オブジェクト 1 1 3 を生成する。

【0 0 3 9】一実施の形態にあつては、各コピー・オペレータの実行は、編集リスト 1 0 8 の中で識別されたすべての編集要求を迅速に処理する並行フォーマットの多重処理ユニットによって処理することができる。さらに、編集リストの中にはきまった評価順序は存在せず、また各編集操作は独立に行なうことができるので、並行処理は容易である。他の一実施の形態にあつては、インターネット・ビデオ・サーバーを用いて多重処理を行なうことができる。当業者には周知のように、インターネット・ビデオ・サーバーは、編集リスト 1 0 8 の中の編集要求を同時に処理するために用いることができる。

【0 0 4 0】やはり図 2 を参照して、編集リスト 1 0 8 の中の各コピー要求のために適当なグル・ファイルが生成されたら、M E D I T エンジン 1 0 2 は、第二のパスで編集リスト 1 0 8 を通読し、編集リスト 1 0 8 の中で識別された各チャンネルのためにスティッチャ・オブジェクト 1 4 7 および 1 4 8 を生成する。図示の例では、チャンネル 1 およびチャンネル N のために生成された二つのスティッチャ・オブジェクトのみが示されているが、編集リスト 1 0 8 の中で識別されたチャンネルの数に応じて任意の数のスティッチャ・オブジェクトを生成できることは理解されよう。例として、実施の形態によっては、M P E G - 2 プラットホームの下の約 4 0 0 0 のビデオ・チャンネルおよび約 8 0 0 0 のオーディオ・チャンネルの多重チャンネルのためのスティッチャ・オブジェクトを含むようにすることができるものもある。

【0 0 4 1】各チャンネルのためにスティッチャ・オブジェクトが生成されると、各スティッチャ・オブジェクト 1 4 7 および 1 4 8 は、グル・オブジェクト 1 3 0 お

よび 1 3 1 を生成することが好ましい。この実施の形態にあつては、各スティッチャ・オブジェクトは、編集リストを通読して関連するチャンネルのための編集要求をさがす。例として、スティッチャ・オブジェクト 1 4 7 は、編集リスト 1 0 8 を通読してチャンネル 1 のための編集要求を識別し、同様に、スティッチャ・オブジェクト 1 4 8 は、編集リスト 1 0 8 を通読してチャンネル N のための編集オペレータを識別する等々の構成とされる。グル・オブジェクト 1 3 0 および 1 3 1 が生成されると、グル・オブジェクト 1 3 0 は、各スティッチャ・オブジェクト 1 4 7 および 1 4 8 に、最初のパスの間に生成されたグル・データを提供する。

【0 0 4 2】この例では、グル・オブジェクト 1 3 0 は、コピーされたセグメントのために各種のグル・セグメントを引き出す任務をもつ。例として、グル・オブジェクト 1 3 0 は、A. M P E G グル・ファイル 1 2 6 の中に記憶されたグル・データを引き出してそれをスティッチャ・オブジェクト 1 4 7 に提供するものとするのもできる。さらに、なんらかのミドル・グルデータ（すなわち、切り取られたセグメントの未処理部分）が要求された場合、グル・オブジェクト 1 3 0 は、制御オブジェクト 1 1 1 によって制御されるストリーマ 1 2 2 へのポインタ 1 3 4 を用いる。このようにして、グル・オブジェクト 1 3 0 は、A. M P E G ファイル 1 2 4 から正しいフレームを引き出すことができる。この実施の形態にあつては、ミドル・グルは、図 1 の表示順序ストリーム 5 2 内のフレーム 1 5 から 2 3 に関連させることができる。したがって、各スティッチャ・オブジェクト 1 4 7 および 1 4 8 は、グル・データを要求し、グル・オブジェクト 1 3 0 および 1 3 1 は、適当な位置からデータを引き出す。各スティッチャ・オブジェクトが、時系列的に要求されたデータを引き出すと、各スティッチャ・オブジェクトは、P E S データ・ストリームを M U X ユニット 1 5 0 へ転送し、該ユニットは、引き出された P E S データ・ストリームを多重化して、単一のストリームを M E D I T 1 0 2 を介してアプリケーション 1 0 6 へ送る。

【0 0 4 3】図 3 は、本発明の一実施の形態にもとづいてビデオ・ファイル内部でターゲット・フレームを検索することに関係するステップを示すフローチャートである。この方法では、ステップ 6 0 0 でビデオ・ファイルのフレーム・レートを求める。一般に、ビデオ・ファイルのフレーム・レートは、多重ファイルのビデオ部分のシーケンスヘッダ、とくに最初のビデオ・パケットのヘッダから求められる。上に述べたように、ビデオ・ファイルのフレーム・レートは、N T S C では毎秒約 3 0 ビデオ・フレームであり、P A L では毎秒約 2 5 ビデオ・フレームである。

【0 0 4 4】ステップ 6 0 0 でビデオ・ファイルのフレーム・レートを求めたら、ステップ 6 0 2 へ進み、ビデ

オおよびオーディオをともに含むファイルのビット・レートを求める。当業者には周知のように、ビット・レートは、多重ストリームの中の最初のパックのバックヘッダ（例、MUXビット・レート）から求められる。ステップ602で1秒あたりのビット数としてのビット・レートが求められたら、ステップ604へ進み、ビデオ・ファイルの終わりから「バイト」であらわしたビデオ・ファイルの大きさが判別される。例として、適当な操作システムを用いれば、通常は、バイトであらわした読み取りファイルの大きさを確認することができる。したがって、ステップ604では、好ましくは、オペレーティング・システムによって多重ビデオ／オーディオ・ファイルの大きさがあたえられる。

【0045】ステップ604でビデオ／オーディオ・ファイルの大きさが判別されたら、ステップ606へ進み、ビデオ・ファイルの全フレーム数の推定が行なわれる。例として、フレーム数は、ファイルの大きさにフレーム・レートを「掛け」、その積をビット・レートで割ることによって推定することができる。

【0046】表1

フレーム数の推定値

S=ファイルの大きさ（バイト数）

B=ビット・レート（バイト数／秒）

R=フレーム・レート（フレーム数／秒）

T=全フレーム数の推定値（フレーム）

$T = S \times R / B$

ステップ606で上に示した式にもとづいて全フレーム数の推定値を求めたら、ステップ608へ進み、ビデオ／オーディオ・ファイル内部のターゲット・ビデオ・フレームのバイト数であらわした時間位置を推定する。ビデオ・ファイル内部のターゲット・ビデオ・フレームの時間位置を推定するためには、ターゲット・ビデオ・フレーム番号にビット・レートを掛けて、フレーム・レートで割る。次に、求めた時間位置から1秒のバイト数を引いて、ターゲット・フレームの実際の位置の前にあるある位置に到達するようにする。

【0047】表2

ターゲット・ビデオ・フレームの推定時間位置（ETP）

S=ファイルの大きさ（バイト数）

B=ビット・レート（バイト数／秒）

R=フレーム・レート（フレーム数／秒）

T=全フレーム数の推定値（フレーム）

F=ターゲット・フレーム番号

$ETP = F \times S / T \equiv F \times B / R - B \equiv B (F - 1) / R$

ビデオ・ターゲット・フレームの推定時間位置（ETP）は、ステップ608で上に示した式を用いて求められることは理解されよう。処理は、次にステップ610へ進み、次のGOPヘッダを読み取って、一時メモリ、

例えばキャッシュ・メモリあるいは他の適当な記憶媒体に保管する。例として、次のGOPヘッダは、推定時間位置のすぐ後のGOPヘッダであることが好ましい。ステップ610で次のGOPヘッダが保管されたら、処理は、決定ステップ612へ進み、次のGOPヘッダ時間コードが、ターゲット・フレーム番号より大きいフレーム番号を示すか否かが判別される。

【0048】例として、ターゲット・フレーム番号が、フレーム番号5525であり、GOPヘッダ時間コードが、フレーム番号5450を示したとすると、示されたフレーム番号は、ターゲット・フレーム番号より小さいことになる。当業者には周知のように、GOPヘッダが読み取られると、該GOPヘッダの中のSMPTE時間コードを読み取ることによってフレーム番号が示される場合がある。MPEGの資料に示されているように、GOPヘッダに含まれるSMPTE時間コードを読み取って復号するために、任意の適当な周知のアルゴリズムを用いることができる。より詳しくは、MPEG資料「動画および関連オーディオの符号化」—約1.5Mビット／秒までのデジタル記憶媒体用（第2部）、2-付録E、IEC規格、刊行物461、第二版、「ビデオ・テープ・レコーダー用時間および制御コード」（1986）（MPEG document "Coding of Moving Pictures and Associated Audio"—For digital storage media at up to about 1.5 Mbits/s (Part 2), 2-annex E, IEC Standard, Publication 461, second edition, entitled "Time and Control Code For Video Tape Recorders"）を参照されたい。MPEG資料は、すべて、参考資料として本出願に添付されている。

【0049】したがって、次のGOPヘッダ時間コードがターゲット・フレーム番号より大きくないフレーム番号を示すと判別された場合には、処理は、ステップ610へ戻って、上に述べたように次のGOPヘッダが読み取られて保管される。次に、処理は、再びステップ612へ進んで、次のGOPヘッダ番号がターゲット・フレーム番号より大きいフレーム番号を示すか否かが再び判別される。他方、GOPヘッダ時間コードが、ターゲット・フレーム番号より大きいフレーム番号を示すと判別された場合には、処理は、ステップ614へ進む。

【0050】ステップ614では、処理が1GOPヘッダだけ逆戻りし、GOPヘッダを「ターゲットGOPヘッダ」として識別する。すなわち、ターゲットGOPヘッダは、ターゲット・フレーム番号を含むGOPを画定するものである。さらに、前に読み取られた各GOPヘッダは保管されているので、1フレームの逆戻りは、単に処理を逆転させて保管されているGOPへ戻ることにしか過ぎない。このようにして、GOPヘッダ時間コードは、ターゲット・フレーム自身のフレーム番号かまたは少なくともターゲット・フレーム番号より大きくないフレーム番号を示すことになる。

【0051】ステップ614で処理が1GOPヘッダだけ逆戻りしてGOPヘッダをターゲットGOPヘッダとした後、処理は、ステップ616へ進み、ターゲット・フレーム番号を検索するためにターゲットGOP内部のあらかじめ定められた数のフレームを読み取る。あらかじめ定められたターゲット・フレームの数が読み取られると、ターゲット・フレームが識別され、処理が完了する。

【0052】図4は、ターゲットGOP内部であらかじめ定められた数のフレームを読み取ってターゲット・フレーム番号の検索を実行することに関係する処理ステップをより詳細に示した図である。処理は、ステップ618から始まり、このステップでは、ターゲット画像グループ(GOP)内部の次のフレームの画像ヘッダが読み取られる。上に述べたように、GOPヘッダは、一般に、1フレームで始まるある数のビデオ・フレームの始まりを画定する。ターゲットGOP内部の最初のフレームのための画像ヘッダが読み取られたら、処理は、決定ステップ620へ進み、現在のフレームの時間表示番号にターゲットGOPヘッダ(例、SMPTE時間コード)から得られるフレーム番号を加えたものがターゲット・フレーム番号に等しいか否かが判別される。

【0053】現在の時間基準フレーム番号にターゲットGOPヘッダから得られるフレーム番号を加算したものがターゲット・フレーム番号に等しくないと判別された場合には、処理は、再びステップ618へ進み、次のフレームの画像ヘッダが読み取られて復号される。ステップ618で次のフレームの画像ヘッダが読み取られて復号されると、処理は、再び決定ステップ620へ進む。決定ステップ620では、もう一度、現在の時間基準フレーム番号にターゲットGOPヘッダから得られるフレーム番号を加えたものがターゲット・フレーム番号に等しいか否かが判別される。この条件が満たされる場合には、処理は、ステップ622へ進んでターゲット・フレームが識別され、ターゲット・フレームの検索を実行する処理が完了する。

【0054】ターゲット・フレームが識別されると、当該ターゲット・フレームのファイル・バイト・オフセットがわかり、記憶される。本明細書で使用する限りにおいて、ファイル・バイト・オフセットという用語は、ビデオ・ファイルの中でのファイルの始めに対するターゲット・フレームの位置として定義される。すなわち、ターゲット・フレームは、「0」バイト(すなわち、オフセットなし)に設定されるファイルの始めから判別されたバイト数である「ファイル・バイト・オフセット」だけファイル内に入り込んだ位置に位置づめることができる。ターゲット・フレームに関するファイル・オフセットが知られているため、必要な場合には、単にターゲット・フレームの記憶されているファイル・バイト・オフセットを求めるだけで、ターゲット・フレームの内容

に迅速にアクセスしてその内容を読み取ることができる。

【0055】図5は、例として、本発明の一実施の形態の始点と終点をもつビデオ・ファイル702を示す線図である。図3のステップ608で説明したように、検索エンジンは、最初、飛び越しによってビデオ/オーディオ・ファイルの中の推定時間位置まで進む。図5に示すように、ビデオ・ファイルの中の推定位置への飛び越しが完了すると、GOPヘッダ710から得られるフレーム番号がターゲット・フレーム(例、ターゲット・フレーム5525)より大きいかが判別される。この例では、GOPヘッダ710は、ターゲット・フレーム5525より大きいビデオ・フレーム番号を示さない。

【0056】したがって、処理は、次のGOPヘッダ712へ進み、再び、GOPヘッダ時間コードが例として挙げたターゲット・フレーム5525より大きいフレーム番号を示すか否かが判別される。説明のために、GOPヘッダ712から得られるフレーム番号が、再び、ビデオ・ターゲット・フレーム番号5525より大きくないと判別されたとする。処理は、もう一度次のGOPヘッダ714へ進むが、このGOPヘッダも、ターゲット・フレームより大きいフレーム番号を示さない。処理は、さらに次のGOPヘッダ716へ進む。この時点で、ついに、GOPヘッダ時間コードが、ターゲット・ビデオ・フレーム5525より大きいフレーム番号を示すと判別される。本発明の一実施の形態にあつては、図3のステップ614で説明したように、処理は、前の保管GOPヘッダ714へ逆戻りする。

【0057】図5の拡大した704に示すように、GOPヘッダ714の後には多数のビデオ・フレームが続いている。すなわち、これら複数のビデオ・フレームが、画像グループを画定し、この画像グループは、1フレーム5523(時間基準フレーム番号2)、B-フレーム5521(時間基準フレーム番号0)、B-フレーム5522(時間基準フレーム番号1)、P-フレーム5526(時間基準フレーム番号5)、B-フレーム5524(時間基準フレーム番号3)、およびB-フレーム5525(時間基準フレーム番号4)を有する。MPEG規格に記されているように、GOPヘッダ714のSMPTE時間コードから得られるフレーム番号は、矢印で示すB-フレーム5521である。これらさまざまな検索操作が行なわれている間、ビデオ・ファイルは、「符号化順序」にしたがっていることが理解されよう。しかし、各種の時間基準フレーム番号によって、適当な「表示順序」のどれかが識別されることになる。

【0058】GOPヘッダ714から得られるフレーム番号は、フレーム番号5521であるため、検索エンジンは、ターゲット・フレーム5525から4フレームだけ離れていることになる。したがって、検索エンジンは、4番目の時間基準フレーム番号を識別する処理へ進

む。この例では、4番目の時間基準フレーム番号は、「ターゲット・フレーム5525」である。図5の拡大した704に示すように、ターゲット・フレーム5525は、B-フレームである。

【0059】図6Aは、本発明の一実施の形態における多重ビデオ／オーディオ・ファイル内のビデオ・フレーム数を効率的に判別するための処理ステップを示すフローチャートである。処理は、ステップ800で始まり、このステップでは、ビデオおよびオーディオ・ファイルの終わりがバイト数で識別される。上に述べたように、一般的に、任意の適当なオペレーティング・システムにおいては、通常のファイルを読み取り、その終わりを長さとして判別することができる。ステップ800でビデオ・ファイルの終わりが判別されると、処理は、ステップ802へ進み、検索エンジンは、バイト数で1秒だけ逆戻りする。説明をわかり易くするために、図6Bには、図6Aのフローチャートを用いて説明する開始時間と終了時間をもつファイル850の例を示してある。

【0060】上に述べたように、検索エンジンは、好ましくは、例として示したファイル850の終わりからバイト数で1秒逆戻りして、点851に達する。検索エンジンがバイト数で1秒逆戻りすると、処理は、ステップ804へ進み、次のGOPヘッダが読み取られて保管される。例として、ファイル850では、次のGOPヘッダは、GOPヘッダ854として示されている。

【0061】次に、処理は、決定ステップ806へ進み、検索エンジンが前方へ移動して、ビデオ・ファイルの中に次のGOPヘッダがあるか否かを判別する。例として示したファイル850は次のGOPヘッダ856を含んでいるため、検索エンジンは、次のGOPヘッダ856へ進み、このGOPヘッダが、図6Bに示すように適当に読み取られて保管される。次に、処理は、再び決定ステップ806へ進み、検索エンジンが前方へ移動して、ビデオ・ファイルの中に次のGOPヘッダがあるか否かを判別する。

【0062】ビデオ・ファイルの中にはもうGOPヘッダはないので、処理は、ステップ808へ進み、検索エンジンは、前に保管したGOPヘッダ856へ逆戻りする。この時点で、検索エンジンは、ファイルの中の最後のGOPヘッダを識別している。したがって、ファイルの中の最後のビデオ・フレームはGOP856内部に置かれている。図6Bに示すように、検索エンジンが前に保管したGOPヘッダ856へくると、処理は、ステップ810へ進み、GOPヘッダ856に続く各フレームの画像ヘッダが読み取られる。このようにして、読み取られた各画像ヘッダの時間基準フレーム番号が判別されて保管される。

【0063】図6Bの例として示したファイル850では、次のフレームは、時間基準フレーム番号「2」をもつ「I」フレーム15, 523である。この番号は、一

時的にメモリに保管される。処理は、ステップ812へ進み、次のフレームの画像ヘッダが読み取られ、時間基準フレーム番号が判別される。図6Bに示すように、次のフレームは、時間基準フレーム番号「0」をもつ

「B」フレーム15, 521であり、この番号は、保管される。この時点で、「前の」時間基準フレーム番号「2」と「現在の」時間基準フレーム番号「0」が保管されたことになる。

【0064】次に、処理は、決定ステップ814へ進み、「現在の」時間基準フレーム番号「0」が「前の」時間基準フレーム番号「2」より大きいかが判別される。現在の時間基準フレーム番号「0」は前の時間基準フレーム番号「2」より大きくないので、処理は、ステップ815へ進み、他の画像ヘッダがあるか否かが判別される。他の画像ヘッダは存在しないので、処理は、ステップ816へ進み、現在の時間基準フレーム番号「0」をもつフレームは、無視される。

【0065】処理は、再びステップ812へ進み、次の画像ヘッダが読み取られ、その時間基準フレーム番号が判別される。次のフレームは、時間基準フレーム番号「1」をもつ「B」フレームである。この時点で、「現在の」時間基準フレーム番号は「1」であり、「前の」時間基準フレーム番号は「2」である。ここで、処理は、決定ステップ814へ進み、「現在の」時間基準フレーム番号「1」が前の時間基準フレーム番号「2」より大きいかが判別される。現在の時間基準フレーム番号は前の時間基準フレーム番号より大きくないので、処理は、ステップ815へ進み、他の画像ヘッダがあるか否かが判別される。

【0066】他の画像ヘッダは存在しないので、処理は、ステップ816へ進む。上に述べたように、現在の時間基準フレーム番号は無視され、処理は、再びステップ812へ進む。ステップ812では、時間基準フレーム番号「5」をもつ「P」フレーム15, 526の画像ヘッダが読み取られる。この時点で、「現在の」時間基準フレーム番号は「5」であり、「前の」時間基準フレーム番号は「2」である。

【0067】ここで再び、処理は、決定ステップ814へ進み、「現在の」時間基準フレーム番号「5」が前の時間基準フレーム番号「2」より大きいかが判別される。現在の時間基準フレーム番号は前の時間基準フレーム番号より大きいので、処理は、ステップ817へ進み、前の時間基準フレーム番号「2」をもつフレームは無視される。即ち、現在の時間基準フレーム番号「5」が前の時間基準フレーム番号「2」より大きいため、検索エンジンは、ファイルの中の最後に表示されるフレームの識別により近づくことになる。ファイルの中の最後に表示されるフレームが、ファイル内部に含まれるビデオ・フレームの正確な合計数を示すものであることはいうまでもない。ファイルの中の最後に表示可能なフレー

ムは、「P」フレーム 15, 526 であるが、検索エンジンは、上に述べたように、残るフレームを読み取ってこのことを確認しなければならない。

【0068】したがって、処理は、決定ステップ 818 へ進み、他にも画像ヘッダが存在するか否かが判別される。他にも存在するため、処理は、ステップ 812 へ進み、次の画像ヘッダが読み取られ、時間基準フレーム番号が判別される。図 6 B に示すように、次のフレームは、時間基準フレーム番号「3」をもつ「B」フレーム 15, 524 である。したがって、「現在の」時間基準フレーム番号は「3」であり、「前の」時間基準フレーム番号は「5」である。

【0069】ここで、処理は、決定ステップ 814 へ進み、「現在の」時間基準フレーム番号「3」が前の時間基準フレーム番号「5」より大きいかが否かが判別される。この場合、現在の時間基準フレーム番号「3」は前の時間基準フレーム番号「5」より大きくないので、処理は、ステップ 815 へ進み、他の画像ヘッダがあるかが判別される。他の画像ヘッダは存在するので、処理は、ステップ 816 へ進み、現在の時間基準フレーム番号「3」をもつフレームは、無視される。処理は、次にステップ 812 へ進み、次の画像ヘッダが読み取られ、その時間基準フレーム番号が判別される。

【0070】図 6 B に示すように、次のフレームは、時間基準フレーム番号「4」をもつ「B」フレームである。したがって、「現在の」時間基準フレーム番号は「4」であり、「前の」時間基準フレーム番号は「5」である。ここで、処理は、決定ステップ 814 へ進み、現在の時間基準フレーム番号が前の時間基準フレーム番号より大きいかが否かが判別される。この場合、現在の時間基準フレーム番号「4」は前の時間基準フレーム番号「5」より大きくない。次に、処理は、ステップ 815 へ進み、他の画像ヘッダがあるかが否かが判別される。

【0071】図 6 B の例では、「B」フレーム 15, 525 の後には画像ヘッダは存在しないので、処理は、ステップ 819 へ進み、現在の時間基準フレーム番号「4」をもつフレームは、無視される。この時点で、残る唯一のフレームは、時間基準フレーム番号「5」をもつ「P」フレーム 15, 526 である。これで、検索エンジンは、時間基準フレーム番号が最も大きい画像ヘッダをもつフレームを確認し、それより小さい時間基準フレーム番号をもつフレームはすべて無視したことになる。

【0072】処理は、次に、ステップ 820 へ進み、識別された時間基準フレーム番号が、GOP ヘッダ SMPTE 時間コードから判別されたフレーム番号に加えられる。上に述べたように、MPEG 資料に示されておりまた参考資料として本出願に添付した周知のアルゴリズムを用いて、GOP ヘッダ SMPTE から絶対フレーム番号を判別することができる。MPEG 資料に示されてい

るように、得られる絶対フレーム番号は、「B」フレーム 15, 521 である。

【0073】フレームが識別されたら、それが、上に識別した時間基準フレーム番号「5」に加えられる。即ち、最後のフレーム番号は、 $(15, 521 + 5 = 15, 526)$ である。この時点で、検索エンジンは、「P」フレームをフレーム番号「15, 526」をもつ最後のフレームとして正しく識別したことになる。検索エンジンは、ファイルの中の各フレームを手間をかけて読み取って復号する必要なしに、図 6 B に例として示したファイル 850 の中のビデオ・フレームの数を正確に判別したことになる。

【0074】図 7 は、本発明の一実施の形態における、複数のオーディオおよびビデオ・パケットをもつシステム・ストリーム 900 を示す線図である。図 7 は、「システム・クロック」のシステム・ストリーム 900 内部でターゲット・ビデオ・フレーム 918 の位置を判別するための図 8 に記した処理ステップを視覚的に示す図である。ターゲット・ビデオ・フレーム 918 の位置が判別されれば、最も近いオーディオ・フレームを識別することができ、それによって「オーディオ・ビデオ」検索が完了する。

【0075】MPEG 資料に記載されているように、ビデオおよびオーディオ構成要素がともに確実に同期化されるようにするためのタイミング機構が配設される。一般的に、MPEG 規格は、システム・クロック基準 (SCR) および同期を維持した適当な再生を確保するために 33 ビットを用いて符号化された再生タイムスタンプ (PTS) をともに識別する。さらに、システム・クロックは、一般に、約 90 kHz で作動する。

【0076】完全さのために、図 7 は、システム・ストリームの左端に位置するオーディオ・パケット 902、および、それに続くビデオ・パケット 904、その後のビデオ・パケット 906、その後の他のビデオ・パケット 908、およびその後のオーディオ・パケット 910 を示す。また、ビデオ・パケット 904 を拡大して、パケット・ヘッダ 911、複数の画像ヘッダ 912、914、および 916 を識別したものも示されている。この例では、画像ヘッダ 916 が、ターゲット・ビデオ・フレーム 918 の画像ヘッダである。

【0077】図 8 のフローチャートに示すオーディオ・ビデオ検索を行なうための処理は、ステップ 950 で始まり、このステップで、ターゲット・ビデオ・フレーム 918 の画像ヘッダ 916 とビデオ・パケット・ヘッダ 911 の間に配置されている画像ヘッダの数が判別される。この例では、画像ヘッダの数は、「2」であると判別される。すなわち、画像ヘッダ 912 および 914 である。もちろん、ターゲット・ビデオ・フレーム画像ヘッダ 916 とビデオ・パケット・ヘッダ 911 の間に配置されている画像ヘッダがない場合には、画像ヘッダの

数は、「0」である。

【0078】次に、処理は、ステップ952へ進み、図7のバケット・ヘッダ911から「フレームあたりのシステム・クロック」であらわしたビデオ・バケットタイムスタンプが読み取られる。「フレームあたりのシステム・クロック」としてのビデオ・バケットタイムスタンプは、任意のビデオ・フレーム周波数を用いて判別することができる。例として、30フレームのフレーム周波数を用いた場合には、フレームあたりのシステム・クロックは、下の表から判別することができる。

【0079】表3

フレームあたりのシステム・クロック

90,000 (システム・クロック数/秒) / 30
(フレーム/秒) = 3,000 (システム・クロック/フレーム)

次に、処理は、ステップ954へ進み、「ビデオ・フレームあたりのシステム・クロック」(すなわち、3,000)にターゲット・ビデオ・フレーム画像ヘッダ916とビデオ・バケット・ヘッダ911の間に配置されている画像ヘッダの数が乗算される。ステップ950で判別されたように、この例では、「2」つの画像ヘッダが配置されている。したがって、フレームあたりの3,000システムクロックに2フレームが乗算され、6,000システム・クロックの値が得られる。次に、バケットタイムスタンプに関するクロック数が6,000システム・クロックに加算され、システム・クロックであらわしたターゲット・ビデオ・フレーム画像ヘッダ916の位置が示される。この時点で、検索エンジンは、例として示した図7のファイル900内部のどこに、システム・クロックであらわしたターゲット・ビデオ・フレーム918が存在するかをすでに判別している。

【0080】他方、ステップ950で、ターゲット・ビデオ・フレーム画像ヘッダ916とビデオ・バケット・ヘッダ911の間に配置されている画像ヘッダが存在しないと判別されると、フレームあたり3,000のシステム・クロックに「0」が乗算され、システム・クロックの数は0となる。したがって、ターゲット・ビデオ・フレーム画像ヘッダ916のためのシステム・クロックは、「バケットタイムスタンプ」自身に関するシステム・クロックのみとなる。この時点で、検索エンジンは、バケット・ヘッダ911とターゲット・ビデオ・フレーム画像ヘッダ916の間に存在する画像ヘッダのない特殊な場合においても、例として示した図7のファイル900内部のどこに、システム・クロックであらわしたターゲット・ビデオ・フレーム918が存在するかをすでに判別している。

【0081】ターゲット・ビデオ・フレーム918のシステム・クロックであらわした位置が判別されると、処理は、ステップ956へ進み、判別されたシステム・クロックの合計が一時メモリに保管される。あるいは、適

当なポインタを用いて、システム・クロックであらわしたターゲット・フレームの位置を示すこともできる。次に、処理は、ステップ958へ進み、システム・クロックであらわしてターゲット・ビデオ・フレーム918に最も近いオーディオ・フレームを含むオーディオ・バケットが識別される。例として、最も近いオーディオ・バケットは、最も先行するオーディオ・バケットであることが好ましい。図7の例では、これは、オーディオ・バケット902である。ただし、他の実施の形態にあっては、最も近いオーディオ・フレームは、例えばオーディオ・バケット910のような、その後に続くオーディオ・バケットにあるとすることもできる。

【0082】ステップ958で適当なオーディオ・バケットが識別されると、処理は、ステップ960へ進み、ビデオ・バケットについて上に説明したとほぼ同様にして適当なオーディオ・バケット内での検索が行なわれる。もちろん、選ばれたオーディオ・バケット内部でのオーディオ・フレームの適当な位置を求めるためには、特定のMPEGオーディオ規格、オーディオ・フレーム・レートおよびオーディオ・ビット・レートが用いられる。この時点で、「オーディオ・ビデオ」検索を行なう処理は完了する。

【0083】本発明では、コンピュータ・システムに記憶されたデータを用いるさまざまなコンピュータが実行する操作が用いられる。これらの操作は、物理量の物理的処理を必要とする操作である。通常、これらの量は、記憶、転送、組み合わせ、比較、その他の処理が可能な電気信号または磁気信号の形をとるが、必ずしもそれに限定されるものではない。さらに、行なわれる処理は、生成、識別、判別、または比較などと呼ばれる場合が多い。

【0084】本明細書に記載されまた本発明の一部をなす操作は、すべて、有用なマシン・オペレーションである。本発明は、また、これらの操作を行なうための装置に関するものである。装置は、必要な目的の達成のために特別につくることができるし、あるいは、コンピュータに記憶されているコンピュータ・プログラムによって選択的に活性化されあるいは構成される汎用コンピュータとすることもできる。とくに、本発明の開示内容にもとづいて書かれたコンピュータ・プログラムには、さまざまな汎用マシンを用いることもできるし、あるいは、必要な操作を行なうためのより専門的な装置をつくるのが好便な場合もある。本発明の構成例を以下に示す。

【0085】図9は、本発明にもとづく処理を行なうためのコンピュータ・システム1300の例を示すブロック線図である。コンピュータ・システム1300は、デジタル・コンピュータ1302、表示画面(モニター)1304、プリンタ1306、フロッピー・ディスク・ドライブ1308、ハード・ディスク・ドライブ1310、ネットワーク・インターフェース1312、および

10

20

30

40

50

キーボード 1314 を含む。デジタル・コンピュータ 1302 は、マイクロプロセッサ 1316、メモリ・バス 1318、ランダム・アクセス・メモリ (RAM) 1320、読み取り専用メモリ (ROM) 1322、周辺バス 1324、キーボード・コントローラ 1326 を含む。デジタル・コンピュータ 1302 は、パーソナル・コンピュータ (例えば、IBM コンパティブルなパーソナル・コンピュータ、マッキントッシュ・コンピュータ、またはマッキントッシュ・コンピュータとコンパティブルなコンピュータ)、ワークステーション・コンピュータ (例えば、サン・マイクロシステムズまたはヒューレット・パッカードのワークステーション)、あるいは他の種類のコンピュータとすることができる。

【0086】マイクロプロセッサ 1316 は、汎用デジタル・プロセッサで、コンピュータ・システム 1300 の操作を制御する。マイクロプロセッサ 1316 は、1 チップのプロセッサとすることもできるし、あるいは多数の構成要素で実装することもできる。マイクロプロセッサ 1316 は、メモリから検索された命令を用いて、入力データの受信と処理、および出力装置へのデータの出力と表示を制御する。本発明に関しては、マイクロプロセッサ 1316 の特定の機能として、MPEG ビデオおよびオーディオ・ストリーム内部での検索に関係する処理を補助する機能を挙げることができる。

【0087】メモリ・バス 1318 は、マイクロプロセッサ 1316 が RAM 1320 および ROM 1322 にアクセスするために使用する。RAM 1320 は、マイクロプロセッサ 1316 が一般記憶域としてまたスクラッチ・パッド・メモリとして使用し、また、入力データおよび処理済みデータを記憶するために使用することもできる。ROM 1322 は、マイクロプロセッサ 1316 が実行する命令およびプログラム・コードならびに他のデータを記憶するために用いることができる。

【0088】周辺バス 1324 は、デジタル・コンピュータ 1302 が入力、出力、および記憶装置にアクセスするために用いられる。記載の実施の形態にあっては、これらの装置は、表示画面 1304、プリンタ装置 1306、フロッピー・ディスク・ドライブ 1308、ハード・ディスク・ドライブ 1310、およびネットワーク・インターフェース 1312 を含む。キーボード・コントローラ 1326 は、キーボード 1314 から入力を受け取り、押された各キーの復号されたシンボルをバス 1328 を介してマイクロプロセッサ 1316 へ送るために用いられる。

【0089】表示画面 1304 は、マイクロプロセッサ 1316 によって周辺バス 1324 を介して供給されるか、またはコンピュータ・システム 1300 の他の構成要素によって供給されるデータの映像を表示する出力装置である。プリンタ装置 1306 は、プリンタとして作動する場合には、紙などの上に映像を出力する。プリン

タ装置 1306 の代わりにあるいはそれに加えて、プロッター、タイプセッター等々の他の出力装置も使用することができる。

【0090】フロッピー・ディスク・ドライブ 1308 およびハード・ディスク・ドライブ 1310 は、各種のデータを記憶するために用いることができる。フロッピー・ディスク・ドライブ 1308 は、各種データの他のコンピュータ・システムへの移送を容易にし、ハード・ディスク・ドライブ 1310 は、記憶されている大量のデータへの高速アクセスを可能にする。

【0091】マイクロプロセッサ 1316 は、オペレーティング・システムと組み合わされて、コンピュータ・コードを実行し、データを生成しまた使用する。これらのコンピュータ・コードおよびデータは、RAM 1320、ROM 1322、またはハード・ディスク・ドライブ 1310 に常駐することができる。コンピュータ・コードおよびデータは、また、取りはずし自在のプログラム媒体に常駐し、また、必要なときにはコンピュータ・システム 1300 にロードまたはインストールすることができる。取りはずし自在のプログラム媒体は、例えば、CD-ROM、PC-CARD、フロッピー・ディスク、および磁気テープを含む。

【0092】ネットワーク・インターフェース 1312 は、他のコンピュータ・システムに接続されたネットワークを介してデータを送受信するために用いられる。インターフェース・カードまたは類似の装置およびマイクロプロセッサ 1316 によって実装された適当なソフトウェアを用いれば、コンピュータ・システム 1300 を既存のネットワークへ接続し、標準プロトコルにしたがってデータを転送することができる。

【0093】キーボード 1314 は、ユーザーがこれを用いてコマンドおよび他の命令をコンピュータ・システムへ入力するものである。本発明に関連して他の種類の入力装置を使用することもできる。例えば、コンピュータ・マウス、トラック・ボール、スタイラス (尖筆)、またはタブレットなどの指示具を用いて、汎用コンピュータの画面上のポインタを操作することもできる。

【0094】本発明は、コンピュータが読み取り可能な媒体上のコンピュータが読み取り可能なコードとして実施することもできる。コンピュータが読み取り可能な媒体とは、データを記憶することができ、そのデータを後にコンピュータ・システムで読み取ることのできる任意のデータ記憶装置を意味する。コンピュータが読み取り可能な媒体としては、例として、読み取り専用メモリ、ランダム・アクセス・メモリ、CD-ROM、磁気テープ、光学データ記憶装置を含む。コンピュータが読み取り可能な媒体は、また、ネットワークで連結された複数のコンピュータ・システムに分散させて、コンピュータが読み取り可能なコードが分散的に記憶され実行されるようにすることもできる。

【0095】上に説明したMPEGオーディオおよびビデオ規格で、参考資料として本出願に添付するものは以下の通りである。すなわち、(1)「動画および関連するオーディオ情報の総称的符号化：ビデオ」、ISO/IEC 13818-2 ("Generic Coding of Moving Pictures and Associated Audio Information: Video," ISO/IEC 13818-2)と題する文書、(2)「デジタル記憶媒体のための、約1.5 Mビット/秒までの動画および関連するオーディオの符号化」(第1部システム、第2部ビデオ、第3部オーディオ) 11171/11172 (1995/1996) ("Coding of Moving Pictures and Associated Audio for Digital Storage Media at up to about 1.5 MBit/s" (Part 1 System, Part 2 Video, Part 3 Audio) 11171/11172 (1995/1996))、と題する文書、および(3)「動画および関連するオーディオ情報の総称的符号化」、13818-3 ("Generic Coding of Moving Pictures and Associated Audio Information" ISO/IEC 13818-3)と題する文書である。上に挙げたMPEG規格文書および将来のMPEG規格文書は、すべて、スイス国ジュネーブ20、CH-1211、ISO/IEC私書箱56 (ISO/IEC Case Postale 56, CH-1211, Geneva 20, Switzerland)に依頼すれば、入手可能である。

【0096】以上、本発明の好ましい実施の形態を詳細に説明したが、本発明は、その意図および範囲を逸脱することなく他の形態で実施できるも理解されよう。説明した実施の形態では分散型アーキテクチャーが記載されている。この種のアーキテクチャーは、とくにモジュール構成の面からまた新しい機能の導入の面から多くの効果をもつ。例えば、単に検索エンジン、復号器等々多くの同じ構成要素を利用することのできる追加の「プラグ・イン」オペレータオブジェクトを配設することによって、新しい機能を生成することができる。さらに、上に述べた検索機能は、MPEG以外の規格によって定義されたオーディオビジュアル・ファイル内部でターゲット・フレームを検索し識別するためにも用いることができる。

【0097】上に述べたようなアーキテクチャーは、とくによく機能すると考えられるが、他のアーキテクチャーを用いても同様な機能を得られることは、理解されよう。したがって、上に述べた例および実施の形態は、単に例示であって本発明を制限するものではなく、本発明は、本明細書に記されている詳細に限定されず、添付の特許請求の範囲内で修正が可能であると思料すべきでもある。

【0098】

【発明の効果】本発明は、数多くの効果をもたらすが、とくに大きな効果は、ターゲット・ビデオ・フレームを検索する前あるいはオーディオビジュアル・ファイルのビデオ・フレームの数を判別する前に、手間をかけてオ

ーディオビジュアル・ファイルの中の各ビデオ・フレームを読み取り、指標付けを行なう必要がないことである。

【図面の簡単な説明】

【図1】本発明の一実施の形態にもとづいたビデオ・ファイルの編集に関係する処理ステップを説明するために示した多数のビデオ・フレーム・シーケンスの例である。

【図2】本発明の一実施の形態にもとづいたビデオ・ファイルの編集を示すデータ・フロー・アーキテクチャーの図である。

【図3】本発明の一実施の形態にもとづいたビデオ・ファイル内部でのターゲット・フレームの検索に関係するステップを示すフローチャートである。

【図4】本発明の一実施の形態にもとづいたあらかじめ定められた数のフレームの読み取りに関係する方法のステップを説明する図である。

【図5】本発明の一実施の形態にもとづいたビデオ・ファイルを示す線図である。

【図6A】本発明の一実施の形態にもとづいた多重オーディオビジュアル・ファイルの中のビデオ・フレームの数を効率的に判別するための方法のステップを示すフローチャートである。

【図6B】本発明の一実施の形態にもとづいたビデオ・ファイルの例を示す図である。

【図7】本発明の一実施の形態にもとづいた多数のオーディオおよびビデオ・パケットを有するシステム・ストリームを示す線図である。

【図8】本発明の一実施の形態にもとづいたオーディオ・ビデオ検索を行なうことに関係する方法のステップを示すフローチャートである。

【図9】本発明の一実施の形態にもとづいたオーディオビジュアル編集および検索ステップを行なうためのコンピュータ・システムの例を示すブロック線図である。

【符号の説明】

100…データ・フロー・アーキテクチャー、102…編集エンジン、104…コピー・オペレータ、106…アプリケーション、108…編集リスト、110…チャンネル・オペレータ、111…制御オブジェクト、112…機能オペレータ、113…制御オブジェクト、114…終端オペレータ、115…符号器、116…グル・オブジェクト、118…検索エンジン、120…復号器、122…ストリーマ、124…MPEGファイル、126…MPEGグル・ファイル、130、131…グル・オブジェクト、134…ポインタ、140…記憶媒体、147、148…スティッチャ・オブジェクト、150…マルチプレクサ、702…ビデオ・ファイル、710、712、714、716…GOPヘッダ、850…ファイル、854、856…GOPヘッダ、900…システム・ストリーム、902…オーディオ・パケッ

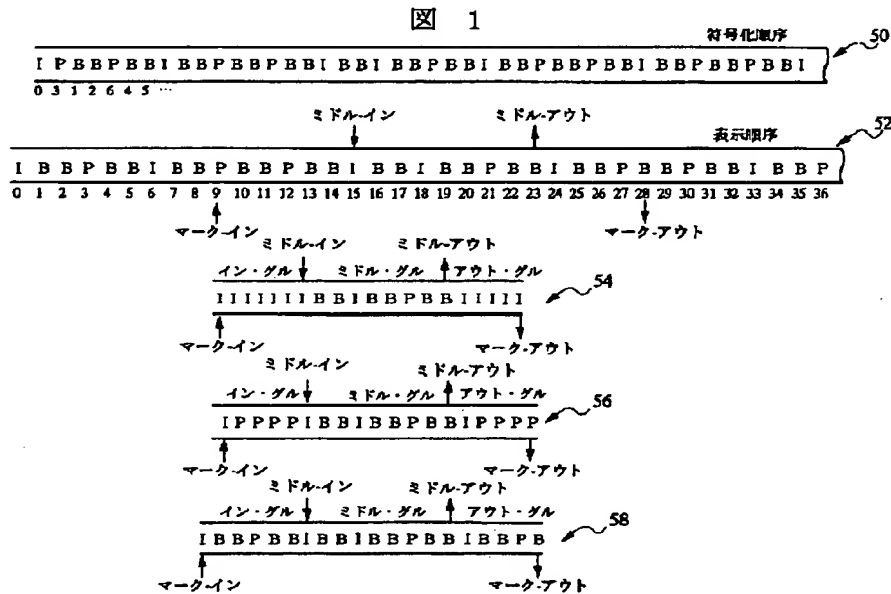
29

ト、904、906、908…ビデオ・パケット、910…オーディオ・パケット、911…パケット・ヘッダ、912、916…画像ヘッダ、918…ターゲット・ビデオ・フレーム、1300…コンピュータ・システム、1302…デジタル・コンピュータ、1304…表示画面、1306…プリンタ、1308…フロッピー・ディスク・ドライブ、1310…ハード・ディスク・ド

30

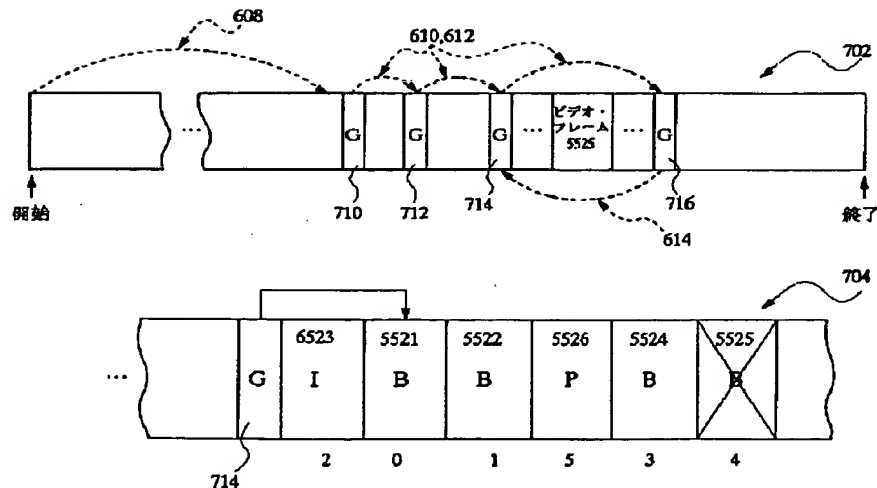
ライブ、1312…ネットワーク・インターフェース、1314…キーボード、1316…マイクロプロセッサ、1318…メモリ・バス、1320…ランダム・アクセス・メモリ、1322…読み取り専用メモリ、1324…周辺バス、1326…キーボード・コントローラ、1328…バス。

【図 1】

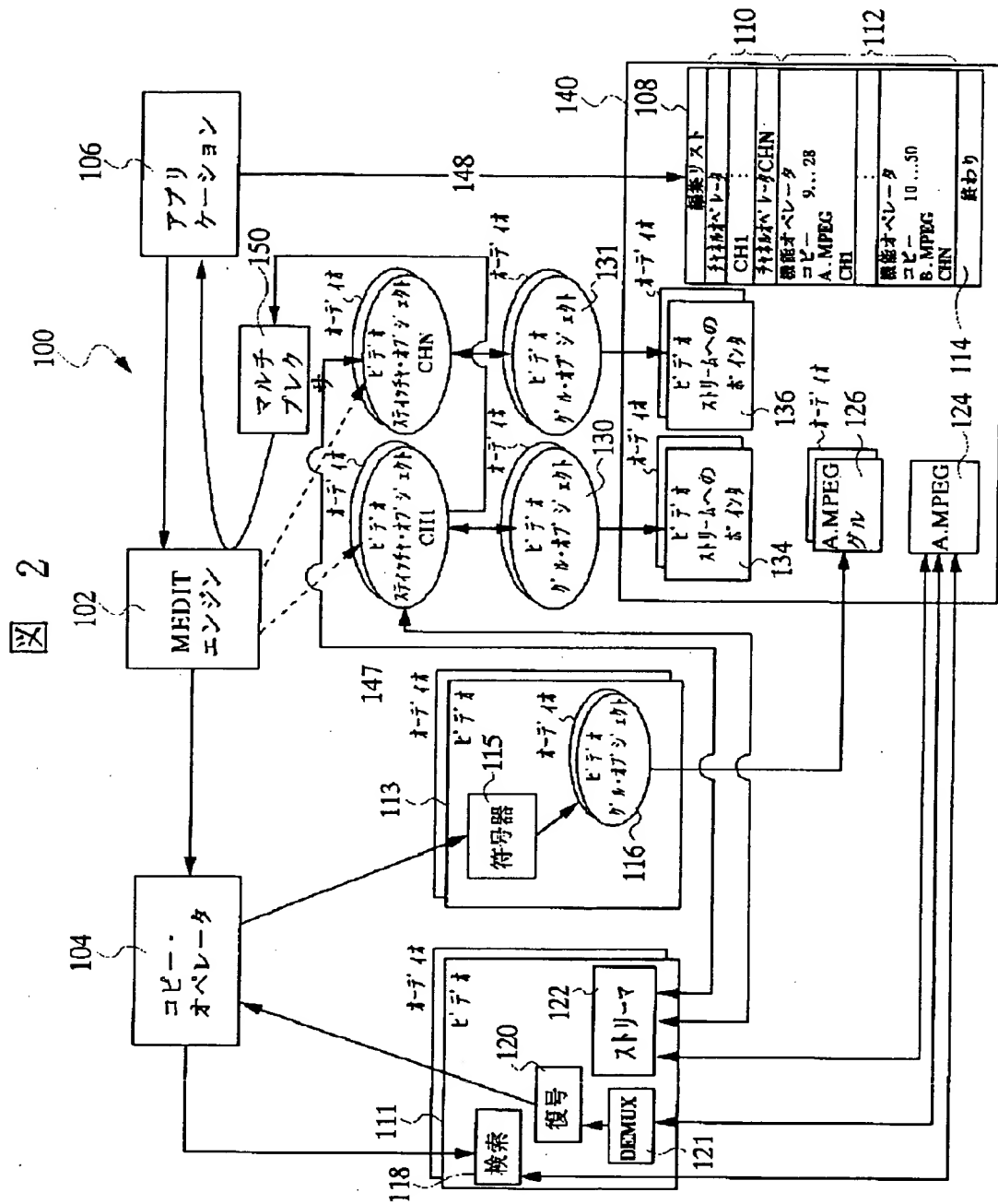


【図 5】

図 5

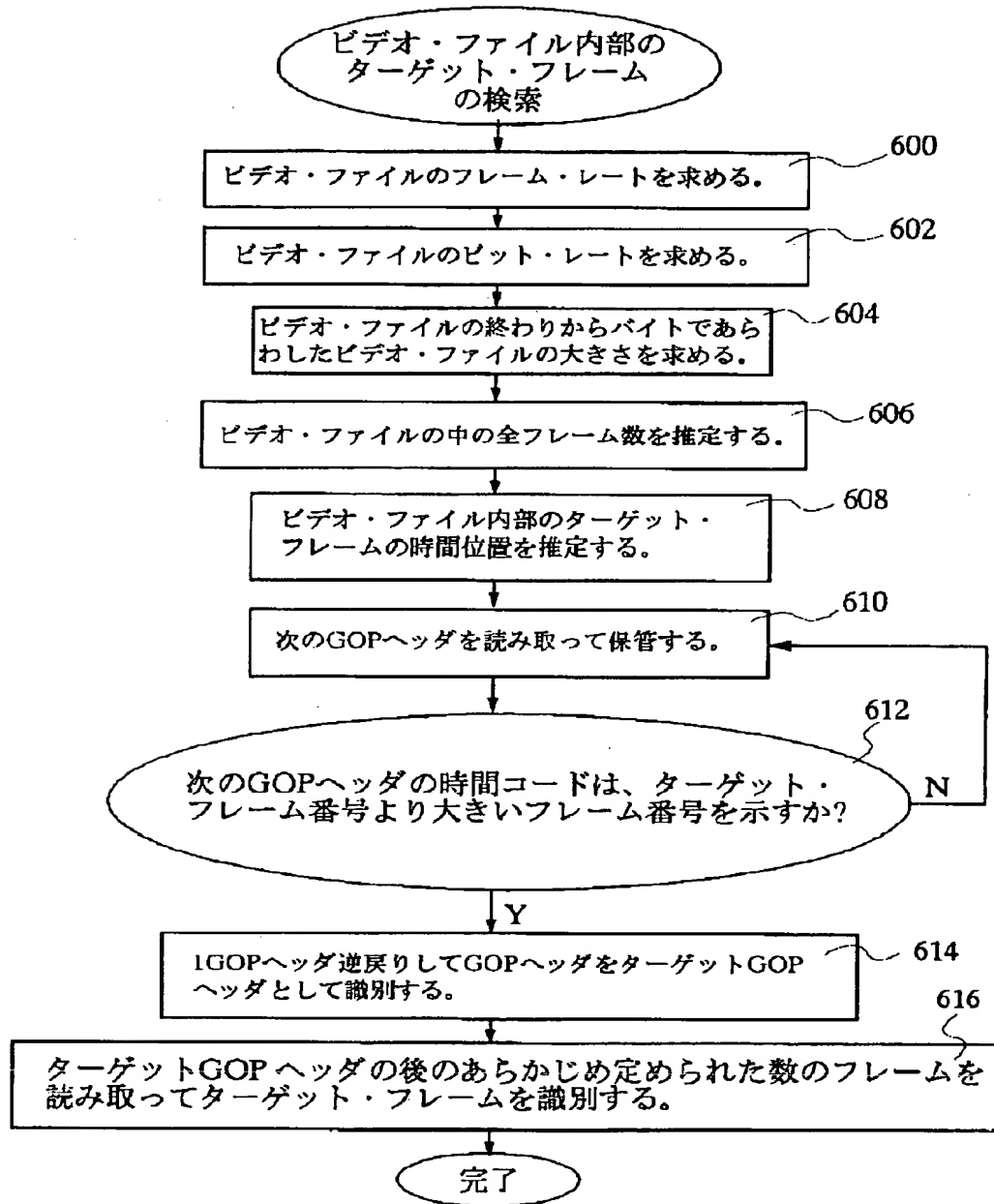


【图 2】

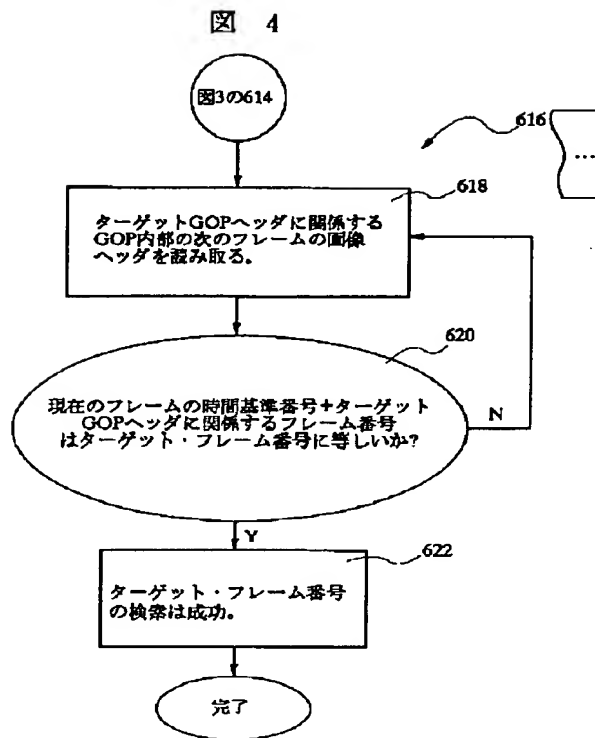


【図 3】

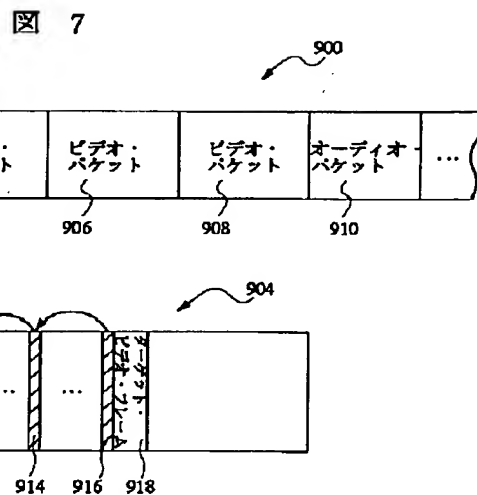
図 3



【図 4】

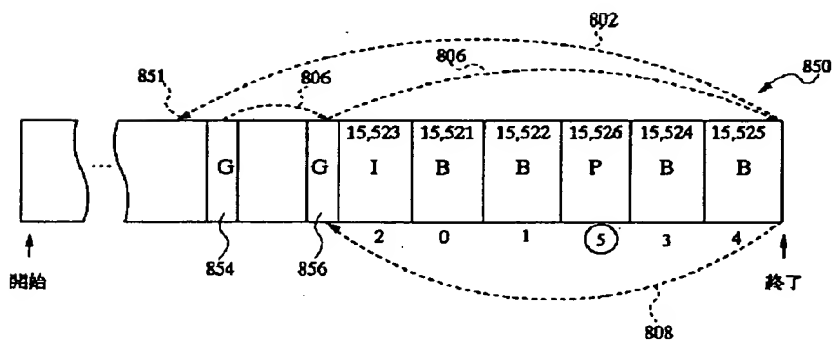


【図 7】



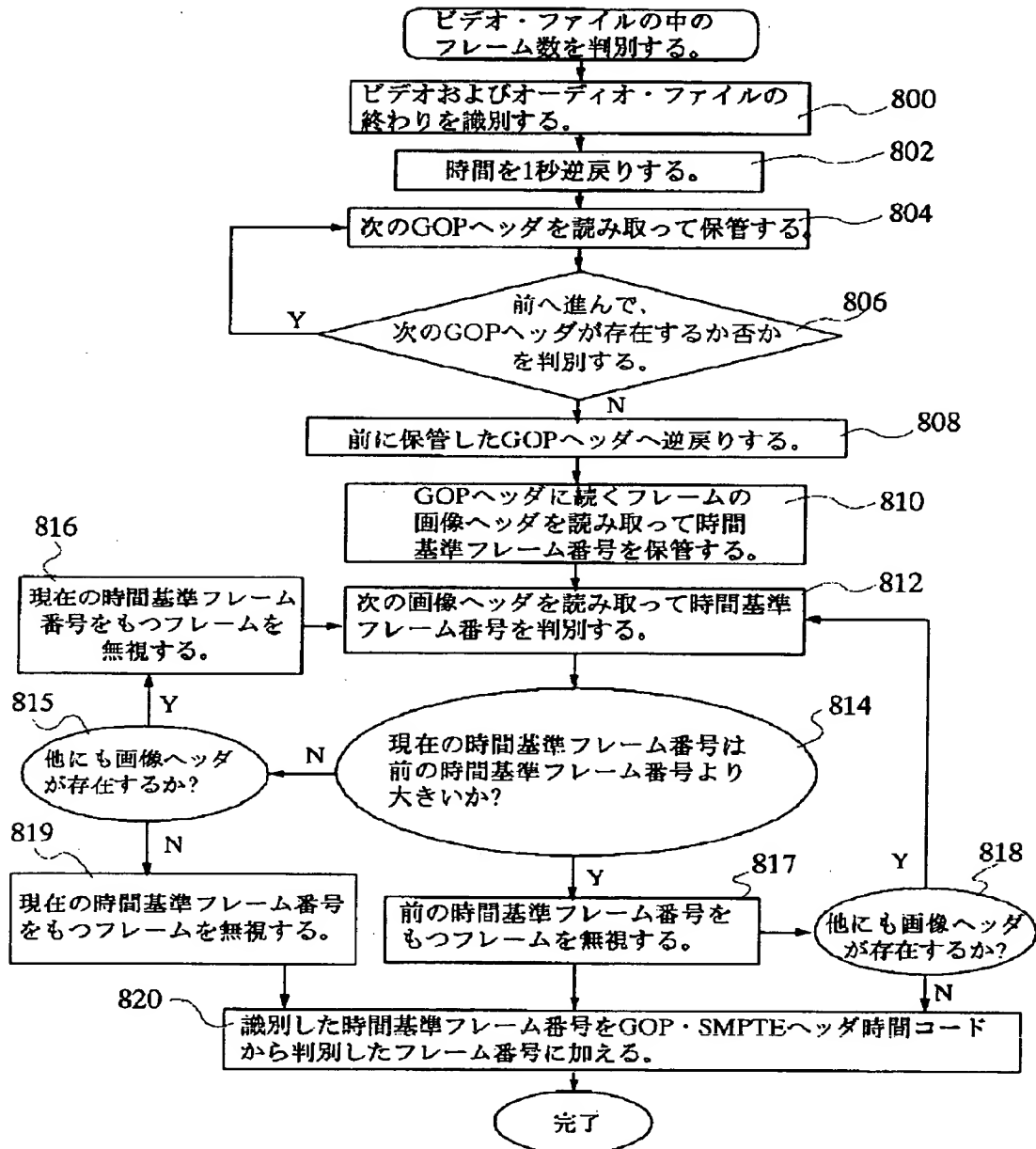
【図 6 B】

図 6B



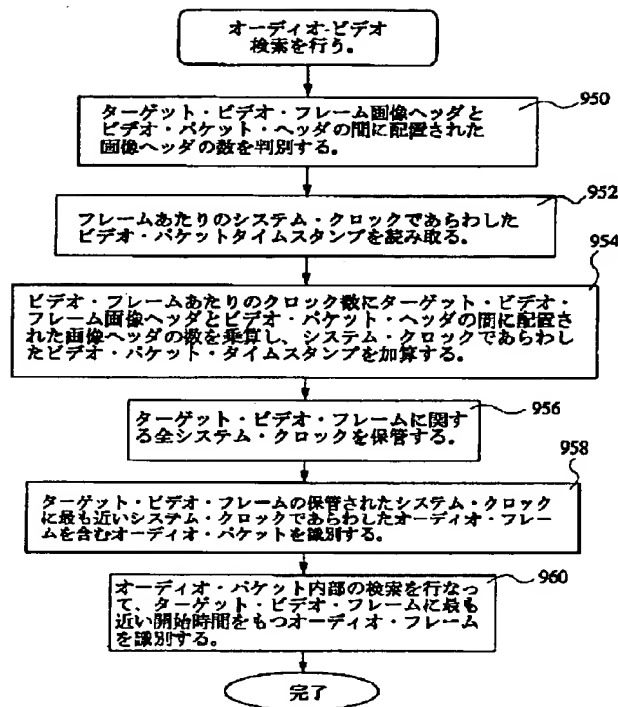
【図 6 A】

図 6A



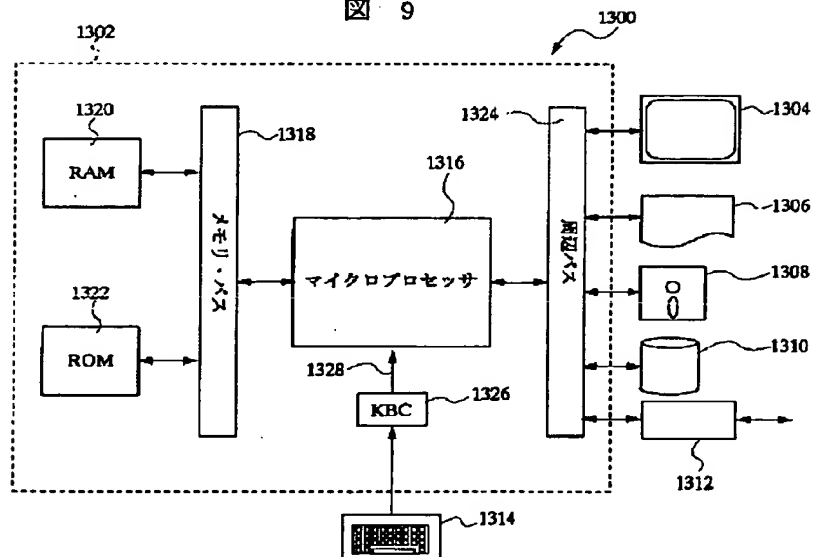
【図 8】

図 8



【図 9】

図 9



フロントページの続き

(72)発明者 アンドリュー・ボールフリーマン
アメリカ合衆国、94086 カリフォル
ニア州、サニーヴェイル、イー・エヴリン
・アヴェニュー 825、アパートメント
456